Monte Carlo Anti-Aliasing

TR88-018

April 1988

John H. Halton

N00014-86-K-0680

The University of North Carolina at Chapel Hill
Department of Computer Science
CB#3175, Sitterson Hall
Chapel Hill, NC 27599-3175

MONTE CARLO ANTI-ALIASING

by

John H. Halton

The University of North Carolina at Chapel Hill
Department of Computer Science
Sitterson Hall, CB#3175
Chapel Hill, NC 27599 U.S.A.

ABSTRACT

Several anti-aliasing strategies are proposed, which generate Monte Carlo discretized estimates of color and intensity at each pixel of a raster display.

MONTE CARLO ANTI ALIASING by JOHN H. HALTON

1. We are given a function $f\colon \mathbf{R}^2 \to \mathbf{R}^1$, specifying color and intensity at any point of a *screen* area $S \subseteq \mathbf{R}^2$. The screen S is subdivided into N *pixels* P_h ($h = 1, 2, \ldots, N$), all disjoint and of equal area and shape.

2. It is intended to approximate the function f on S by a function $\phi\colon \mathbf{R}^2 \to \mathbf{R}^1$ which takes the value ϕ_h on the pixel P_h, for $h = 1, 2, \ldots, N$.

3. One approach is to define, for the pixel P_h centered at $\mathbf{c}_h$, a *weight function* $w(\mathbf{r} - \mathbf{c}_h) = w_h(\mathbf{r})$ and let

$$\phi_h = \int_Q d\mathbf{r}\, f(\mathbf{r}) w_h(\mathbf{r}), \tag{1}$$

where Q denotes $\mathbf{R}^2$ and $\int_Q d\mathbf{r}$ denotes $\int_{-\infty}^{\infty} dx \int_{-\infty}^{\infty} dy$, with $\mathbf{r} = (x, y)$.

4. A very general *Monte Carlo* scheme for estimating ϕ_h would select an integer n_h and a set of estimator-probability pairs $(g_{hi}(\mathbf{r}), \rho_{hi}(\mathbf{r}))$, for $i = 1, 2, \ldots, n_h$; so that one samples points $\boldsymbol{\xi}_i \in Q$ with probability density $\rho_{hi}(\boldsymbol{\xi}_i)$, independently of each-other, and uses the estimator

$$\theta_h = \sum_{i=1}^{n_h} g_{hi}(\boldsymbol{\xi}_i) \tag{2}$$

for ϕ_h. For example, "crude Monte Carlo" could define $\rho_{hi}(\mathbf{r}) = N/A$, where A is the area of S (so that A/N is the area of the pixel P_h), and use the estimator $g_{hi}(\mathbf{r}) = cf(\mathbf{r})$ in P_h; but this would *not* work, since we would want that the estimator be *unbiased*, i.e., that

$$\Sigma_{i=1}^{n_h} E[g_{hi}] = \phi_h, \tag{3}$$

and this reduces, by (1), to $c = A\phi_h/N\theta_h n_h$, where

$$\theta_h = \int_{P_h} \mathrm{d}\mathbf{r}\, f(\mathbf{r}), \tag{4}$$

and we would need to know both ρ_h and θ_h to get ρ_h! Another approach is to use $\rho_{hi}(\mathbf{r}) = w_h(\mathbf{r}) = w(\mathbf{r} - \mathbf{c}_h)$ in the whole of Q (though, of course, most of the probability will be in or near P_h), and use the estimator $g_{hi}(\mathbf{r}) = af(\mathbf{r})$; whence the condition (3) reduces to $a = 1/n_h$, provided that the weight function w_h satisfies (as is usual) the normalizing condition

$$\int_Q \mathrm{d}\mathbf{r}\, w_h(\mathbf{r}) = \int_Q \mathrm{d}\mathbf{r}\, w(\mathbf{r} - \mathbf{c}_h) = \int_Q \mathrm{d}\mathbf{r}\, w(\mathbf{r}) = 1. \tag{5}$$

Of course, this condition is not at all unreasonable. Note that we may, yet again, choose, over the whole of Q, $\rho_{hi}(\mathbf{r}) = w'_h(\mathbf{r})$, a different normalized weight function from w_h (for instance, the normal distribution centered at $\mathbf{c}_h$ and with standard deviation of the order of the diameter of a pixel), and then the estimator would be $g_{hi}(\mathbf{r}) = w_h(\mathbf{r})f(\mathbf{r})/w'_h(\mathbf{r})n_h$, as is readily verified, and this is again feasible; so we note the pair:

$$(g_{hi},\ \rho_{hi}) = \left(\frac{w_h(\mathbf{r})f(\mathbf{r})}{w'_h(\mathbf{r})n_h},\ w'_h(\mathbf{r})\right). \tag{6}$$

3. An alternative approach would be to use a form of *stratified sampling*. Note that, in the technique developed above, all n_h estimators are identical and identically distributed. Suppose, instead, that the pixel P_h is dissected into m identical sub-pixels R_{hj}, and that s_j identical estimators $g_{hj}(\mathbf{r})$ are sampled with density $\rho_{hj}(\mathbf{r})$ in Q, where $\rho_{hj}(\mathbf{r}) = \rho(\mathbf{r} - \mathbf{b}_{hj})$ and $\mathbf{b}_{hj}$ is the center of R_{hj}. We then require, by (3), that

$$\sum_{j=1}^{m} s_j \int_Q \mathrm{d}\boldsymbol{r}\ g_{hj}(\boldsymbol{r})\rho_{hj}(\boldsymbol{r}) = \int_Q \mathrm{d}\boldsymbol{r}\ f(\boldsymbol{r})\omega_h(\boldsymbol{r}). \tag{7}$$

As an example, we could choose the function ρ, and then put

$$g_{hj}(\boldsymbol{r}) = \frac{f(\boldsymbol{r})\ \omega(\boldsymbol{r} - \boldsymbol{c}_h)}{ms_j\ \rho(\boldsymbol{r} - \boldsymbol{b}_{hj})}\ ; \tag{8}$$

where we also must have that

$$\sum_{j=1}^{m} s_j = n_h. \tag{9}$$

6. What we must do to make the method efficient is to minimize (or at least diminish) the *variance* of our estimate. Thus, we note that, for the first technique, given by (6), we have

$$\mathrm{var}\Big[\sum_{i=1}^{n_h} g_{hi}\Big] = \sum_{i=1}^{n_h} \mathrm{var}[g_{hi}] = n_h\left\{\int_Q \mathrm{d}\boldsymbol{r}\left(\frac{\omega_h(\boldsymbol{r})f(\boldsymbol{r})}{\omega_h'(\boldsymbol{r})n_h}\right)^2 \omega_h'(\boldsymbol{r}) - \left(\int_Q \mathrm{d}\boldsymbol{r}\ \frac{\omega_h(\boldsymbol{r})f(\boldsymbol{r})}{\omega_h'(\boldsymbol{r})n_h}\ \omega_h'(\boldsymbol{r})\right)^2\right\} = \frac{1}{n_h}(\lambda_h - \phi_h^{\,2}), \tag{10}$$

where

$$\lambda_h = \int_Q \mathrm{d}\boldsymbol{r}\ \frac{[\omega_h(\boldsymbol{r})]^2[f(\boldsymbol{r})]^2}{\omega_h'(\boldsymbol{r})}. \tag{11}$$

For the second technique, given by (8), we similarly get that

$$\mathrm{var}\Big[\sum_{j=1}^{m} s_j g_{hj}\Big] = \sum_{j=1}^{m} s_j\ \mathrm{var}[g_{hj}] = \sum_{j=1}^{m} s_j\left\{\int_Q \mathrm{d}\boldsymbol{r}\left(\frac{f(\boldsymbol{r})\omega(\boldsymbol{r} - \boldsymbol{c}_h)}{ms_j\rho(\boldsymbol{r} - \boldsymbol{b}_{hj})}\right)^2 \rho(\boldsymbol{r} - \boldsymbol{b}_{hj}) - \left(\int_Q \mathrm{d}\boldsymbol{r}\ \frac{f(\boldsymbol{r})\omega(\boldsymbol{r} - \boldsymbol{c}_h)}{ms_j\rho(\boldsymbol{r} - \boldsymbol{b}_{hj})}\ \rho(\boldsymbol{r} - \boldsymbol{b}_{hj})\right)^2\right\}$$

$$= \sum_{j=1}^{m} \frac{1}{m^2 s_j}(\mu_{hj} - \phi_h^{\,2}), \tag{12}$$

where
$$\mu_{hj} = \int_{a} dr \, \frac{[f(r)]^2[w(r - c_h)]^2}{\rho(r - b_{hj})}. \tag{13}$$

7. If we consider the case of (6), (10), and (11), and first assume that f, w, w', and so ϕ_h and λ_h are all given *a priori*; then we may ask how to choose the numbers of function-evaluations n_h by pixels, so as to make all variances the same, given the sum $n = \Sigma_{k=1}^{N} n_k$. The answer is evidently

$$n_h^* = n(\lambda_h - \phi_h^2)/\Sigma_{k=1}^{N}(\lambda_k - \phi_k^2), \tag{14}$$

and the common value of the variance at every pixel is then

$$\mathrm{var}[\Sigma_{i=1}^{n_h^*} g_{hi}] = \Sigma_{k=1}^{N}(\lambda_k - \phi_k^2)/n. \tag{15}$$

In the case of (8), (12), and (13), with f, w, ρ, and so ϕ_h and μ_h given, we similarly see that we can first optimize over the strata in a single pixel; Lagrangian theory shows that

$$s_j^* = n_h(\mu_{hj} - \phi_h^2)^{\frac{1}{2}}/\Sigma_{k=1}^{m}(\mu_{hk} - \phi_h^2)^{\frac{1}{2}} \tag{16}$$

minimizes the variance at P_h to the value

$$\min \mathrm{var}[\Sigma_{j=1}^{m} g_{hj}] = \frac{1}{m^2 n_h}\left(\Sigma_{j=1}^{m}(\mu_{hj} - \phi_h^2)^{\frac{1}{2}}\right)^2. \tag{17}$$

Note that the Cauchy-Schwartz-Bunyakovsky inequality shows that indeed

$$\frac{1}{m^2 n_h}\left(\Sigma_{j=1}^{m}(\mu_{hj} - \phi_h^2)^{\frac{1}{2}}\right)^2 = \frac{1}{m^2 n_h}\,\Sigma_{j=1}^{m}\left(\frac{(\mu_{hj} - \phi_h^2)^{\frac{1}{2}}}{s_j^{\frac{1}{2}}}\, s_j^{\frac{1}{2}}\right)^2$$

$$\leqslant \frac{1}{m^2 n_h}\left(\Sigma_{j=1}^{m}\frac{\mu_{hj} - \phi_h^2}{s_j}\right)\Sigma_{k=1}^{m} s_k, \tag{18}$$

and the right-hand side of the inequality is the general variance (12), by (9); so that (16) does indeed minimize (not maximize or point-of-inflexion) the variance. Now we proceed, as before, to make all the variances (17) the same; yielding that

$$n_h = n\left\{\Sigma_{j=1}^m (\mu_{hj} - \phi_h^2)^{\frac{1}{2}}\right\}^2 / \Sigma_{k=1}^N \left\{\Sigma_{j=1}^m (\mu_{kj} - \phi_k^2)^{\frac{1}{2}}\right\}^2. \qquad (19)$$

This makes the common value of the variance

$$\min \operatorname{var}[\Sigma_{j=1}^m z_{hj}] = \frac{1}{m^2 n} \Sigma_{h=1}^N \left\{\Sigma_{j=1}^m (\mu_{hj} - \phi_h^2)^{\frac{1}{2}}\right\}^2. \qquad (20)$$

8. As a specific example, we may suppose that S is a rectangle

$$S = (0 \leqslant x \leqslant L_1,\ 0 \leqslant y \leqslant L_2); \qquad (21)$$

and that the index h is (h_1, h_2), with $N = N_1 N_2$ and $0 \leqslant h_t < N_t$ $(t = 1, 2)$, so that P_h is the rectangle

$$P_h = P_{h_1 h_2} = \left(\frac{L_1}{N_1} h_1 \leqslant x \leqslant \frac{L_1}{N_1}(h_1 + 1),\ \frac{L_2}{N_2} h_2 \leqslant y \leqslant \frac{L_2}{N_2}(h_2 + 1)\right), \qquad (22)$$

centered at $\mathbf{c}_h = (c_{h1}, c_{h2})$ with $c_{ht} = \dfrac{L_t}{N_t}(h_t + \frac{1}{2})$ $(t = 1, 2)$. (23)

Similarly, we take $j = (j_1, j_2)$, $m = m_1 m_2$, and $0 \leqslant j_t < m_t$ $(t = 1, 2)$, so that R_{hj} is the $(L_1/N_1 m_1 \times L_2/N_2 m_2)$ rectangle centered at

$$\mathbf{b}_{hj} = (b_{hj1}, b_{hj2}) \text{ with } b_{hjt} = \frac{L_t}{N_t m_t}(m_t h_t + j_t + \tfrac{1}{2}) \quad (t = 1, 2). \qquad (24)$$

We may further postulate that both w_h' and ρ_{hj} take the form of the *normal distribution*, with

$$\phi_h'(r) = \frac{1}{2\pi\gamma} \exp(-\{(x - c_{h1})^2 + (y - c_{h2})^2\}/2\gamma), \tag{25}$$

where $$\gamma = (L_1L_2/N_1N_2)\sigma = (A/N)\sigma, \tag{26}$$

and $$\psi_{hj}(r) = \frac{1}{2\pi\beta} \exp(-\{(x - b_{hj1})^2 + (y - b_{hj2})^2\}/2\beta), \tag{27}$$

where $$\beta = (A/Nm_1m_2)\sigma = (A/Nm)\sigma. \tag{28}$$

Here, σ is a constant for the system, related to the weight function w but not to f or to S and its subdivisions.

Then we have that

$$\lambda_h = \frac{A}{N} 2\pi\sigma \int_0^{L_1} dx \int_0^{L_2} dy \; [f(x, y)]^2[w(x - c_{h1}, y - c_{h2})]^2$$
$$\times \exp(\tfrac{N}{A}\{(x - c_{h1})^2 + (y - c_{h2})^2\}/2\sigma) \tag{29}$$

and $$\mu_{hj} = \frac{A}{Nm} 2\pi\sigma \int_0^{L_1} dx \int_0^{L_2} dy \; [f(x, y)]^2[w(x - c_{h1}, y - c_{h2})]^2$$
$$\times \exp(\tfrac{Nm}{A}\{(x - b_{hj1})^2 + (y - b_{hj2})^2\}/2\sigma). \tag{30}$$

9. The strategies investigated here so far are adaptive only insofar as the optimizing numbers of samples (14) and (16) are to be estimated from Monte Carlo estimates of the λ_h and μ_{hj} which can be obtained simultaneously with the estimates of ϕ_h generated by the estimators (6) and (8), respectively. Since only small samples are to be taken, because f is so laborious to get, the relative sample-sizes (14) and (16) will not be very accurately optimal.

Another approach would attempt to perform *importance sampling* by sequentially approximating $f(x, y)\omega_h(x, y)$ with ω_h'. Since ω_h is given and f is experimentally determined (so, also given), we may write $C(x, y)$ for the product. As we accumulate values of C by sampling (initially with an arbitrary distribution), we can form an increasingly accurate picture of the functional dependence of C on (x, y) and model ω_h' on this.

Alternatively, we may do a sequential *correlated sampling* calculation, in which we fix the sampling density arbitrarily, and then use an estimator of the form $\{C(x, y) - \psi(x, y)\}/\omega_h'(x, y) - \int_Q d\mathbf{r}\, \psi(x, y)$, where ψ is the best approximation to C for which the integral on the right is easily computable.

10. Yet another approach which should be empirically investigated is to use an *ordering* of the sampled values of C to indicate where stratification should occur. First, we sample C at a small number of points in each pixel and tabulate C, x, and y, in order of increasing C. If there is a strong correlation of C with x or with y, split the pixel accordingly and sample a few more points. Repeat, if necessary.

Note that the stratification and sampling are done in the whole of Q, not within the pixel or sub-pixel only. This is to conform with the global form of ω. Note also that ω may be given the full theoretical form, and need not be approximated by a normal distribution itself.